John von Neumann Institute for Computing

A Parallel Algebraic Multigrid Preconditioner Using Algebraic Multicolor Ordering for Magnetic Finite Element Analyses

T. Mifune, N. Obata, T. Iwashita, M. Shimasaki

published in

Parallel Computing:
Current & Future Issues of High-End Computing,
Proceedings of the International Conference ParCo 2005,
G.R. Joubert, W.E. Nagel, F.J. Peters, O. Plata, P. Tirado, E. Zapata (Editors),
John von Neumann Institute for Computing, Jülich,
NIC Series, Vol. **33**, ISBN 3-00-017352-8, pp. 237-244, 2006.

http://www.fz-juelich.de/nic-series/volume33

A Parallel Algebraic Multigrid Preconditioner Using Algebraic Multicolor Ordering for Magnetic Finite Element Analyses

T. Mifune[a], N. Obata[a], T. Iwashita[b], and M. Shimasaki[a]

[a]Department of Electrical Engineering, Kyoto University, 6158510 Kyoto, Japan

[b]Academic Center for Computing and Media Studies, Kyoto University, 6068501 Kyoto, Japan

1. Introduction

Recently, finite element (FE) analyses of magnetic fields have played a major role in the design of various electromagnetic machines. In the analyses, most computation time is consumed by the solution of large-scale sparse linear systems of equations derived from FE formulations. Algebraic multigrid (AMG) techniques [6] are known to be good preconditioners that efficiently accelerate the convergence of basic iterative methods for linear systems of equations with sparse matrices.

This paper studies parallel processing of the AMG preconditioner for FE analyses with coloring strategies. In parallelization of the AMG preconditioner, it is important to devise parallelization of the smoother. Since AMG techniques have been developed as black-box multigrid solvers, it is desirable that parallel processing does not destroy the black-box property of the AMG techniques. We propose a parallel Gauss-Seidel (GS) smoother using algebraic multicolor (AMC) ordering and compare it with the coloring strategy we set out in a previous paper [3].

The AMG preconditioner with the AMC ordered GS (AMCGS) smoother perfectly keeps the black-box property and achieves a nearly linear speed-up with respect to multigrid iterations. However, using AMC ordering might cause the deterioration of the convergence of the preconditioned solver, compared with using a sequential GS smoother. The deterioration of the convergence depends on the coloring strategy. Numerical results demonstrate that the new coloring strategy considerably improves the performance of the parallel solver, in a magnetostatic edge-element analysis for a benchmark model.

Moreover, we present new results of the application of a parallel AMG solver to magnetostatic nodal element analysis.

2. Basic Equations and FE Formulations

In this paper, the performances of the parallel AMG preconditioners are evaluated in the two different magnetostatic FE analyses: edge-element and nodal element analyses.

The two analyses deal with the same phenomenon. In static field analysis, the nodal element analysis has a large advantage with respect to the number of unknowns of the linear equations derived from FE formulations. However, edge-element applications are important in practical uses, e.g., in mode analyses of electromagnetic fields.

2.1. Edge-Element Analyses

The basic equation in the finite edge-element analysis is

$$\nabla \times (\nu \nabla \times \boldsymbol{A}) = \boldsymbol{J}_0. \tag{1}$$

Here $\boldsymbol{A}$, $\boldsymbol{J}_0$, and ν are magnetic vector potential, impressed current density, and magnetic reluctivity, respectively.

The approximate solution of the magnetic vector potential is given by

$$\hat{\boldsymbol{A}} = \sum_i u_i \boldsymbol{w}_i^{\mathrm{e}}. \tag{2}$$

Here, $\boldsymbol{w}_i^{\mathrm{e}}$ and u_i denote the edge-element trial functions [4] and the unknowns of the same number as the edges of the FE mesh, respectively.

The Galerkin formulation of (1) leads to linear systems of equations

$$K^{\mathrm{e}}\boldsymbol{u} = \boldsymbol{f}, \tag{3}$$

$$[K^{\mathrm{e}}]_{ij} = \iiint \nu(\nabla \times \boldsymbol{w}_i^{\mathrm{e}}) \cdot (\nabla \times \boldsymbol{w}_j^{\mathrm{e}})\, dV, \tag{4}$$

$$f_i = \iiint J_0 \cdot \boldsymbol{w}_i^{\mathrm{e}}\, dV + \iint \{\boldsymbol{w}_i^{\mathrm{e}} \times (\nu\nabla \times \hat{\boldsymbol{A}})\} \cdot d\boldsymbol{S}, \tag{5}$$

where $\iiint\, dV$ and $\iint\, d\boldsymbol{S}$ denote the volume and surface integrals over the analyzed domain, respectively. The second term of the right-hand-side of (5) is decided by the boundary conditions of the problem.

2.2. Nodal Element Analyses

The basic equation in finite nodal element analysis is written by

$$-\nabla \cdot (\mu\nabla\varphi) = -\nabla \cdot (\mu\boldsymbol{T}_0), \tag{6}$$

where $\mu = 1/\nu$. The magnetic scalar potential and the current vector potential are denoted by φ and $\boldsymbol{T}_0$, respectively.

The approximate solution of φ is given by

$$\hat{\varphi} = \sum_i v_i w_i^{\mathrm{n}}, \tag{7}$$

where, w_i^{n} and v_i denote the nodal element trial functions and the unknowns of the same number as the nodes of the FE mesh, respectively. The linear system of equations to be solved is written by

$$K^{\mathrm{n}}\boldsymbol{v} = \boldsymbol{g}, \tag{8}$$

$$[K^{\mathrm{n}}]_{ij} = \iiint \mu\nabla w_i^{\mathrm{n}} \cdot \nabla w_i^{\mathrm{n}}\, dV, \tag{9}$$

$$g_i = \iiint \boldsymbol{T}_0 \cdot \nabla w_i^{\mathrm{n}}\, dV - \iint \mu w_i^{\mathrm{n}}(\boldsymbol{T}_0 - \nabla\hat{\varphi}) \cdot d\boldsymbol{S}. \tag{10}$$

The second term of the right-hand-side of (10) is decided by the boundary conditions.

3. Algebraic Multigrid Methods

Multigrid (MG) methods are known to be efficient multilevel preconditioners for linear systems of equations arising in FE analyses. The convergence of iterative solvers, e.g. the conjugate gradient (CG) method, are efficiently accelerated by MG preconditioners, utilizing the hierarchy of coarse grids.

Different from geometric MG methods, AMG methods have been developed as black-box multigrid techniques [6]. Therefore, coarse grids are automatically constructed in the AMG algorithm.

In this paper, AMG methods are used as preconditioners for CG methods.

3.1. AMG Preconditioner

The AMG preconditioning is executed by computing approximate solutions of $K^{-1}\boldsymbol{r}$ by a few AMG iterations, in each CG iteration. Here, K and $\boldsymbol{r}$ denote the coefficient matrix and the residual vectors, respectively, of the considered problem.

A two-grid AMG iteration is executed as follows. The approximate solution vector of $K^{-1}\boldsymbol{r}$ is denoted by $\boldsymbol{q}$.

1. The vector $\boldsymbol{q}$ is updated by the pre-smoother. (e.g. a forward GS sweep)

2. The vector $\boldsymbol{q}$ is updated by the coarse grid correction.

3. The vector $\boldsymbol{q}$ is updated by the post-smoother. (e.g. a backward GS sweep)

In most AMG applications, GS methods are used as pre- and post- smoothers. Here, with respect to the sequential computation, the forward GS method is applied as the pre-smoother and the backward GS method as the post-smoother.

The coarse grid correction is described by

$$\boldsymbol{q} \leftarrow \boldsymbol{q} + P(P^T K P)^{-1} P^T (\boldsymbol{r} - K\boldsymbol{q}). \tag{11}$$

P is called the prolongation operator, which defines the coarse grid.

When more than one coarse grid is utilized, the inverse of $P^T K P$ in (11) is approximately computed by the reduction of a two-grid AMG iteration. On the coarsest grid, $(P^T K P)^{-1}$ is solved by a direct method.

3.2. Prolongation for Edge-Element Analyses

A special prolongation operator was developed for magnetostatic analyses using edge-elements in [5]. The shifted coefficient matrix [2] instead of K is used with respect to the preconditioning.

3.3. Prolongation for Nodal Element Analyses

The classical AMG technique [6], which was developed for linear systems of equations with symmetric M matrices, is applied for nodal element analyses. The coefficient matrix K^{n} does not strongly violate the symmetric M property.

4. Parallel AMG Preconditioned Solver

It is easy to parallelize the matrix-vector multiplications and the inner product of two vectors, because each row can be independently computed. Therefore, computations except the preconditioning are easily parallelized in the preconditioned CG solver.

In the parallelization of the AMG preconditioning;

- Multiplications by P and P^T are easily parallelized.

- Since the number of unknowns is small enough on the coarsest grid, computation cost consumed by the direct method is negligible.

- Forward and backward GS smoothers generally include sequential computations, i.e., forward and backward substitutions.

```
for (i = 0 ; i < N ; i++){
  m = 0
  (*)
  for (j = 0 ; j < i ; j++){
    if (K[i][j] != 0 && COLOR[j] == m){
      m++;
      goto (*);
    }
  }
  COLOR[i] = m;
}
```

Figure 1. Algebraic Multicolor Ordering Algorithm 1 (AMC1)

Consequently, it is important to devise parallelization of the smoothers.

Since AMG techniques have been developed as black-box multigrid solvers, which are easily used as library software [6], it is desirable that the parallel processing of the AMG preconditioning does not destroy the black-box property.

In the previous paper, we proposed a parallel AMCGS smoother [3]. Figure 1 shows the coloring algorithm in [3] (AMC1). Here, entry (i, j) of the array K and i-th entry of the array COLOR represents the entry (i, j) of the coefficient matrix and the color number of the i-th unknown, respectively. The AMC coloring strategy guarantees that, if COLOR[i] is equal to COLOR[j], K[i][j] is equal to zero. Because any two unknowns having the same color number can be independently updated in the AMCGS sweep, the AMCGS smoother is efficiently parallelized.

Only the nonzero pattern of the coefficient matrix is utilized in the AMC ordering algorithm. This keeps the black-box property of the AMG preconditioner perfect.

The convergence behavior of the preconditioned solver might change using the AMCGS smoother, compared with using the sequential GS smoother, although the convergence does not depend on the number of processors employed.

In the AMC1 algorithm, the number of colors used is decided by the nonzero pattern of the coefficient matrix; and the number of the colors is less than the maximum of the number of nonzero entries per row.

In this paper we introduce another coloring algorithm [1], which was originally developed for a parallel ICCG solver. Figure 2 shows the coloring algorithm (AMC2). Different from AMC1 ordering, the number of colors “ncolor” in the AMC2 algorithm is set as a parameter before the execution of the algorithm. A selection of the number of colors might improve the convergence of the iterative solver.

5. Numerical Results

Figure 3 shows the sample model, TEAM (Testing Electromagnetic Analysis Methods) benchmark problem 10, which is discretized using tetrahedral elements. The numbers of tetrahedral elements, edges, and nodes are 277874, 333857, and 50082, respectively.

All computations are performed on a shared memory parallel computer, a Fujitsu PRIMEPOWER HPC2500. The Fortran codes are parallelized using the OpenMP directives, using compile option

```
m = 0
for (i = 0 ; i < N ; i++){
  (*)
  for (j = 0 ; j < i ; j++){
    if (K[i][j] != 0 && COLOR[j] == m){
      m = mod(m++, ncolor);
      goto (*);
    }
  }
  COLOR[i] = m;
}
```

Figure 2. Algebraic Multicolor Ordering Algorithm 2 (AMC2)

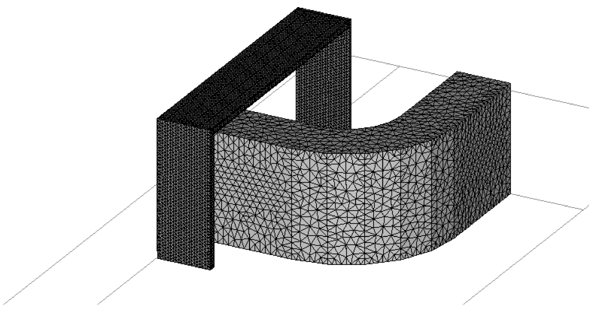

Figure 3. Example of Finite Element Mesh

"-Kfast_GP2=3 -Knolargepage -KOMP -Knoeval -X9."

In the remainder of this paper, the "sequential AMGCG" solver means the AMG preconditioned CG solver using two forward (backward) GS sweeps as the pre- (post-) smoother, which is sequentially executed. The "AMGCG-AMC1GS" mean the AMG preconditioned CG solver using two forward (backward) AMC1GS sweeps as the pre- (post-) smoother. Similarly the "AMGCG-AMC2GS" solver uses two forward (backward) AMC2GS sweeps as the pre- (post-) smoother.

5.1. Edge-Element Analysis

Table 1 presents the number of the sequential AMGCG and AMGCG-AMC1GS iterations, in the edge-element analysis. The number of colors decided by the AMC1 ordering algorithm is about 13, which is nearly constant on all grids.

Table 2 presents the number of AMGCG-AMC2GS iterations when the number of colors is set to various values. The best CG convergence is obtained when the number of colors is 70, although the number of AMGCG-AMC2GS iterations does not greatly change by the selection of the number of colors. The CG convergence, which deteriorates using AMC1 ordering, is significantly improved using the AMC2 algorithm.

Table 3 compares the elapsed time of the AMGCG solvers, in which T_{s} and T_{i} denote the times

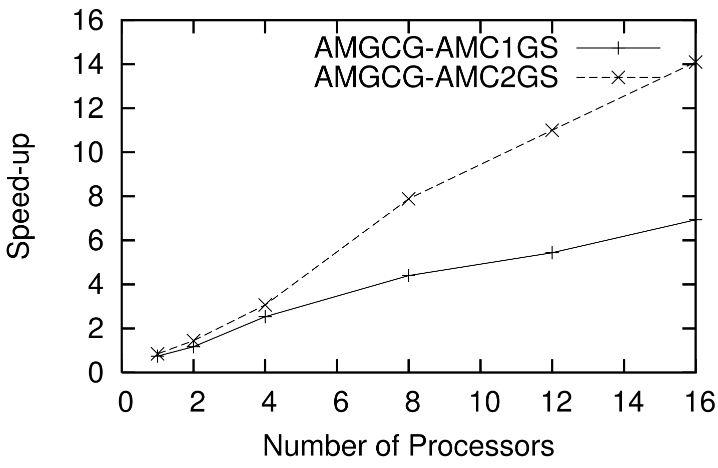

Figure 4. Speed-up of AMGCG Solvers with AMC1 and AMC2 Orderings (Edge-Element Analysis)

Table 1
Number of Sequential AMGCG and AMGCG-AMC1GS Iterations (Edge-Element Analysis)

Sequential AMGCG	AMGCG-AMC1GS
191	289

consumed by the grid construction and by preconditioned CG iterations, respectively. The number of colors of the AMC2 algorithm is set to be 70. With respect to T_{i}, parallel AMGCG solvers attain nearly linear speed-up, due to the property that the number of processors used does not affect the convergence.

Figure 4 demonstrates performances of the parallel solvers, in which the "speed-up" is calculated by $(T_{\mathrm{s}} + T_{\mathrm{i}})_{|\mathrm{Sequential AMGCG}}/(T_{\mathrm{s}} + T_{\mathrm{i}})$. A very good performance is achieved using the AMC2 ordering algorithm.

5.2. Nodal Element Analysis

Table 4 gives the number of sequential AMGCG and AMGCG-AMC1GS iterations, in the nodal element analysis. The maximum number of colors is 31 in the AMC1 algorithm.

Table 5 presents the number of the AMGCG-AMC2GS iterations with respect to various numbers of colors, "ncolor" in Figure 2. The number of AMGCG-AMC2GS iterations does not change in the table.

Table 6 compares the elapsed time. The number of colors of the AMC2 algorithm is set to 60. Parallel solvers achieve good speed-up with respect to T_{i}, although the sequential parts (T_{s}) occupy most of the total time when using 4 processors or more.

Table 2
Number of AMGCG-AMC2GS Iterations (Edge-Element Analysis)

Number of colors	Number of AMGCG-AMC2GS Iterations
60	195
70	193
80	195
90	195

Table 3
Elapsed Time of the AMGCG Solvers (Edge-Element Analysis)

Number of Processors	AMGCG-AMC1GS		AMGCG-AMC2GS		Sequential AMGCG	
	T_s [s]	T_i [s]	T_s [s]	T_i [s]	T_s [s]	T_i [s]
1	5.49	928.22	6.40	816.22	5.40	685.7
2	5.18	585.38	7.23	471.13	-	-
4	4.67	268.90	6.03	218.99	-	-
8	4.22	153.03	5.77	81.91	-	-
16	4.18	95.54	5.54	43.34	-	-

Table 4
Number of Sequential AMGCG Iterations and AMGCG-AMC1GS Iterations (Nodal Element Analysis)

Sequential AMGCG	AMGCG-AMC1GS
17	17

Table 5
Number of AMGCG-AMC2GS Iterations (Nodal Element Analysis)

Number of colors	Number of AMGCG-AMC2GS Iterations
60	17
70	17
80	17
90	17

Table 6
Elapsed Time of the AMGCG Solvers (Nodal Element Analysis)

Number of Processors	AMGCG-AMC1GS		AMGCG-AMC2GS		Sequential AMGCG	
	T_s [s]	T_i [s]	T_s [s]	T_i [s]	T_s [s]	T_i [s]
1	2.79	4.51	3.78	4.66	2.92	4.03
2	3.03	2.83	3.02	2.06	-	-
4	2.88	1.38	3.19	1.27	-	-
8	2.98	0.88	3.29	0.77	-	-
16	2.56	0.58	3.18	0.46	-	-

6. Conclusion

For efficient parallelization of the AMG preconditioner, we introduce an AMC2 algorithm different from that used in previous work. In edge-element magnetostatic analysis, it is demonstrated that the CG convergence is significantly improved using the AMC2 algorithm.

Numerical results using the nodal elements are also presented. The parallel AMG efficiency is not good because the AMG setup process, which is sequentially executed, consumes a large part of the total solution time. However, the speed-up of the iteration part is very good. The parallelization by AMC ordering will be effective in time-dependent nodal element analyses, in which the setup time is negligible.

7. Acknowledgement

Computation in the present paper was carried out as a collaborative research project using the KDK system of the Research Institute for Sustainable Humanosphere (RISH) at Kyoto University.

References

[1] T. Iwashita and M. Shimasaki. Algebraic multicolor ordering for parallelized ICCG solver in finite-element analyses. *IEEE Trans. Magn.*, 38(2):429–432, 2002.

[2] T. Mifune, T. Iwashita, and M. Shimasaki. New algebraic multigrid preconditioning for iterative solvers in electromagnetic finite edge-element analyses. *IEEE Trans. Magn.*, 39(3):1677–1680, 2003.

[3] T. Mifune, T. Iwashita, and M. Shimasaki. A parallel algebraic multigrid solver for fast magnetic edge-element analyses. *IEEE Trans. Magn.*, 41(5):1660–1663, 2005.

[4] J. Nédélec. A new family of mixed finite elements in R^3. *Numer. Math.*, 38:57–81, 1986.

[5] S. Reitzinger and J. Schoberl. An algebraic multigrid method for finite element discretizations with edge elements. *Numer. Linear Algebra Appl.*, 9:223–238, 2002.

[6] J. Ruge and K. Stüben. Algebraic multigrid. In S. McCormick, editor, *Multigrid Methods*, volume 3 of *Frontiers in Applied Mathematics*, pages 73–130. SIAM, Philadelphia, 1987.